

High Level Design & Low Level Design

**Index**

1. Introduction ------------------------------------------------ 3

1.1 Intended audience ------------------------------------------------ 3

1.2 Project purpose ------------------------------------------------ 3

1.3 Key project objective ------------------------------------------------ 3

1.4 Project scope ------------------------------------------------ 4

2. Design overview ------------------------------------------------ 4

2.1 Design objective ------------------------------------------------ 6

2.2 Design alternative ------------------------------------------------ 6

2.3 User interface paradigms ------------------------------------------------ 6

2.4 Validations ------------------------------------------------6

-

3. System architecture ------------------------------------------------ 7

3.1 Database architecture ------------------------------------------------ 7

4. Detailed system design -------------------------------------------------9

4.1 Flowchart of main application ------------------------------------------------9

4.2 Flowchart of maintain database() ------------------------------------------------10

4.3 Flowchart of main menu ------------------------------------------------11

4.4 Flowchart of show report() ------------------------------------------------12

5. Tools Report -------------------------------------------------13

5.1 Strace ------------------------------------------------13

5.2 Valgrind ------------------------------------------------13

5.3 Splint ------------------------------------------------14

5.1 Gcov ------------------------------------------------15

6. Testing ------------------------------------------------24

6.1 Unit Testing ------------------------------------------------24

6.2 Integration Testing —---------------------------------------------25

7. Requirements Traceability Matrix(RTM) ------------------------------------------------30

**1. Introduction**

The bowling scorer application is a system that allows the user to automatically store the scores for each and every player in a systematic order inside the database removing the traditional way of using scorekeeping where every player had to keep their own scores. Now with the introduction of this system it simplifies the process as now every player record is accessible at a single click right from their names to previous tournaments won to how much the player has scored in every single frame throughout the tournament and calculating the winner at the end of the day on the basis of total score of the players.

**1.1 Intended Audience: -**

The target audience set for this project can be identified as an organizer who is conducting a bowling tournament and automate the process of handling the tournament which includes maintaining records of the players and their scores,

**1.2 Project Purpose: -**

The bowling game is a project that helps us understand the basic concepts of functions, file handling, and data structure. The automatic bowling scorer will take bowler id from the bowler. It will create a scoresheet for the entire tournament. It will record all the tournament details and who won in the game. If the bowler is new then it will save in the bowler’s record. We can maintain a database of bowlers by adding, displaying, updating and removing by the main menu of this Bowling Game program.

**1.3 Key Project Objectives: -**

1. Allow the Bowler to enter the Bowler ID
2. Play the game
3. updating the scores for each frame
4. Displays all the records of Bowler
5. Modify/Update the Bowle records.
6. calculating the cumulative score.

**1.4 Project scope : -**

This project aims to create the development of an Automatic Bowling score application,Which takes the bowler information such as ID and Name, adds it to the database and processes the score of the player after the game and adds it to the Bowling data of the user as how many tournaments won along with also displaying the bowling game score of the user to which frame the user is currently bowling in and what is the total score after each frame.

**2. Design Overview: -**

* **Bowling Game comprises of the following modules in maintain bowler database:**

| Name of the Module | Add Module |
| --- | --- |
| Handled by |  |
| Description | The bowler adds the record in the database |

| Name of the Module | Delete Module |
| --- | --- |
| Handled by |  |
| Description | The bowler deletes a record from database |

| Name of the Module | View Module |
| --- | --- |
| Handled by |  |
| Description | The bowler views the record from database |

| Name of the Module | Edit Module |
| --- | --- |
| Handled by |  |
| Description | The bowler edits the record in the database |

* **Bowling Game comprises of the following modules in play the game:**

| Name of the Module | Random Number module |
| --- | --- |
| Handled by |  |
| Description | Generating the random numbers |

| Name of the Module | Strike Condition Module |
| --- | --- |
| Handled by |  |
| Description | Knocking down all the pins with the first ball ends the frame |

| Name of the Module | Spare Condition Module |
| --- | --- |
| Handled by |  |
| Description | Knocking down all the pins with the first two balls |

| Name of the Module | Extra balls Module |
| --- | --- |
| Handled by |  |
| Description | A spare in the tenth frame earns one extra ball |

| Name of the Module | Play the game Module |
| --- | --- |
| Handled by |  |
| Description | Bowler will play the game and awarded with scores |

* **Bowling Game comprises of the following modules in show reports:**

| Name of the Module | Bowler datasheet Module |
| --- | --- |
| Handled by |  |
| Description | It will show the bowler datasheet from bowler database |

| Name of the Module | Bowling day report Module |
| --- | --- |
| Handled by |  |
| Description | It will display the bowling day report from bowling database |

## Design Objectives:

1. Add different bowler profiles to the records.
2. Start the scoring application.
3. Updating the scores for each frame and the total score.
4. Displays all the records of Bowler.
5. Modify/Update the Bowler records
6. Calculating the cumulative score.
   1. **Design Alternative: -**

We have used a linked list structure to store data i.e.. Bowler ID, Name, Years of Experience and Number of Tournaments won, Frame no, balls, score.

### 2.3 User Interface Paradigms: -

The Bowling game provides an option to Bowlers by generating the score automatically in each frame while playing and keeping the records of players also.

### Validation: -

* Bowler Id should not be blank and Duplication is not allowed and characters aren't allowed in the ID.
* In case of integer validation, if the entered Bowler Id is not Integer it displays the message ID must contain Integer only and should not contain more than 15 digits.
* We check for the validity of the name; it should not contain more than 15 characters and Bowler name should not be empty and only alphabets.

**3. SYSTEM ARCHITECTURE: -**

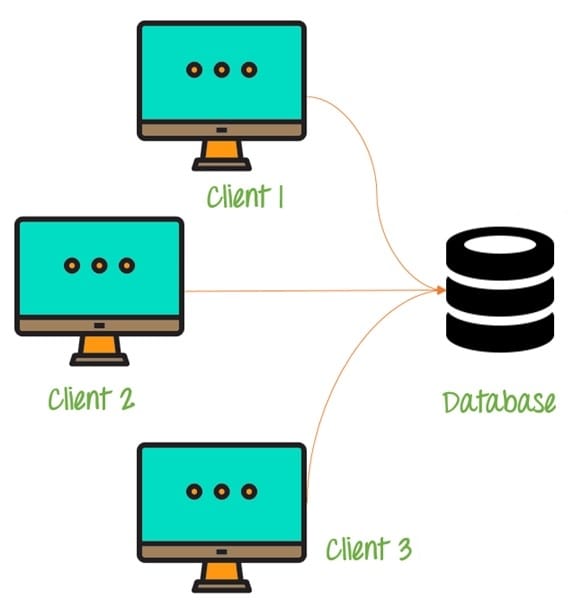
**3.1. Database Architecture**

The architecture used in this system comprises of the database architecture. It is a representation of the database management system design, wherein you can design, develop, implement and maintain the database. This architecture allows dividing the database into different components that can be independently modified, changed, replaced and altered as required for the system.

The database architecture is divided into three tiers namely:-

* 1 - Tier Architecture
* 2 - Tier Architecture
* 3 - Tier Architecture

Our system is based on the Tier 1 model of the database architecture. In this type of model the database is directly available to the user, the user can directly access the database and all of its contents. Which enables the user to directly interact and execute operations.



Some of the characteristics of Database Architecture are:

Self-Describing Nature of a Database System :

* One of the most fundamental characteristics of the database approach is that the database system contains not only the database itself but also an entire definition or description of the database structure and constraints also known as metadata of the database.

Isolation between Data, Programs and Data Abstraction:

* In a traditional file processing system, the structure of database knowledge files is embedded within the application programs, so any changes to the structure of a file may require changing all programs that access that file.

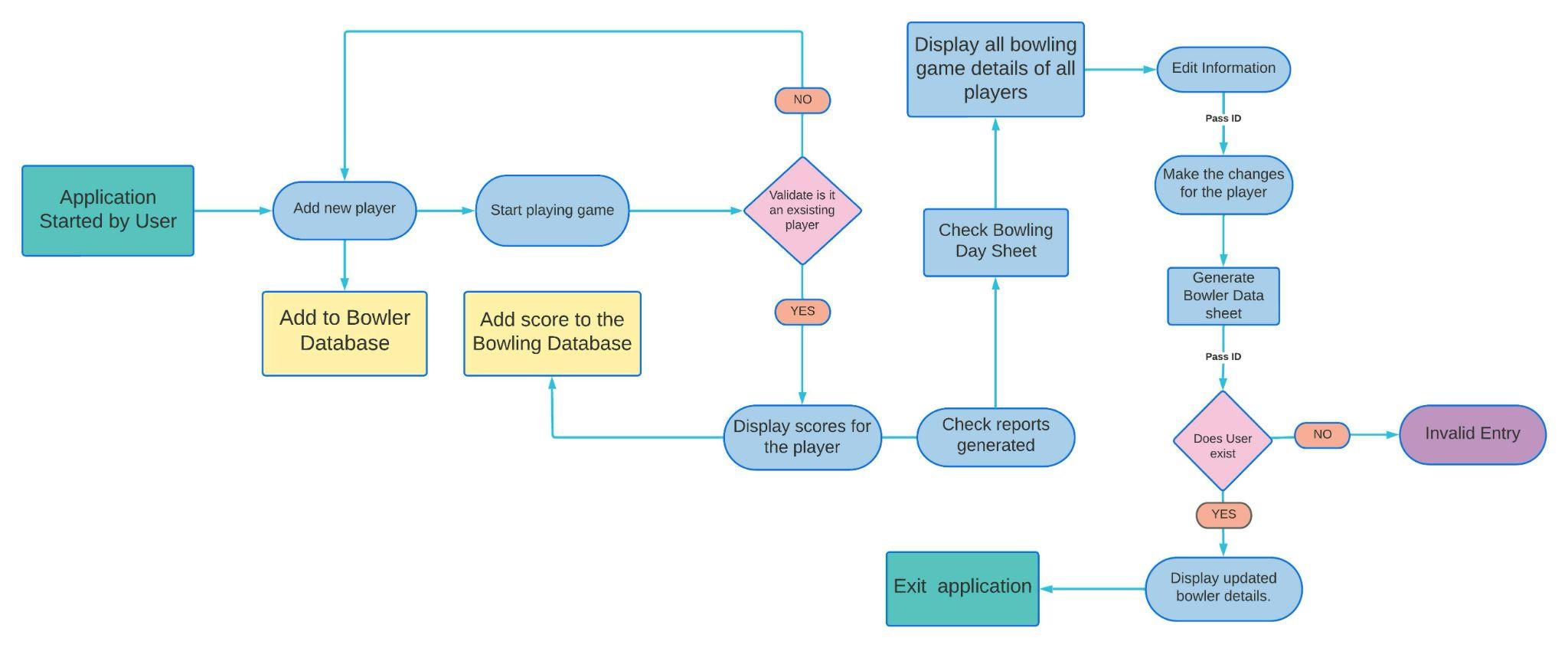
Support for Multiple Views of the Data :

* A database sometimes has many users, each of whom may require a special perspective or view of the database.

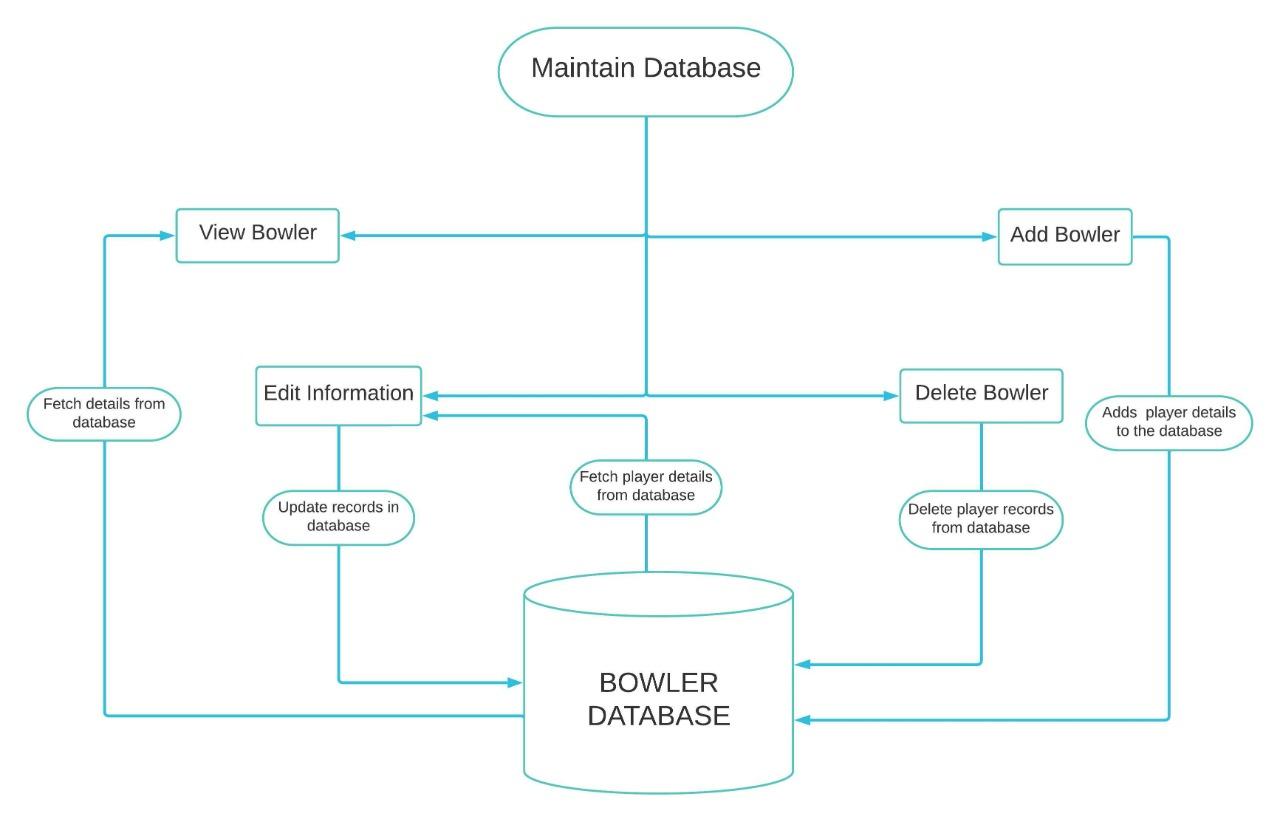
Sharing of knowledge and Multi-user Transaction Processing :

* A multi-user DBMS, as its name implies, must allow multiple users to access the database at an equivalent time or concurrently.

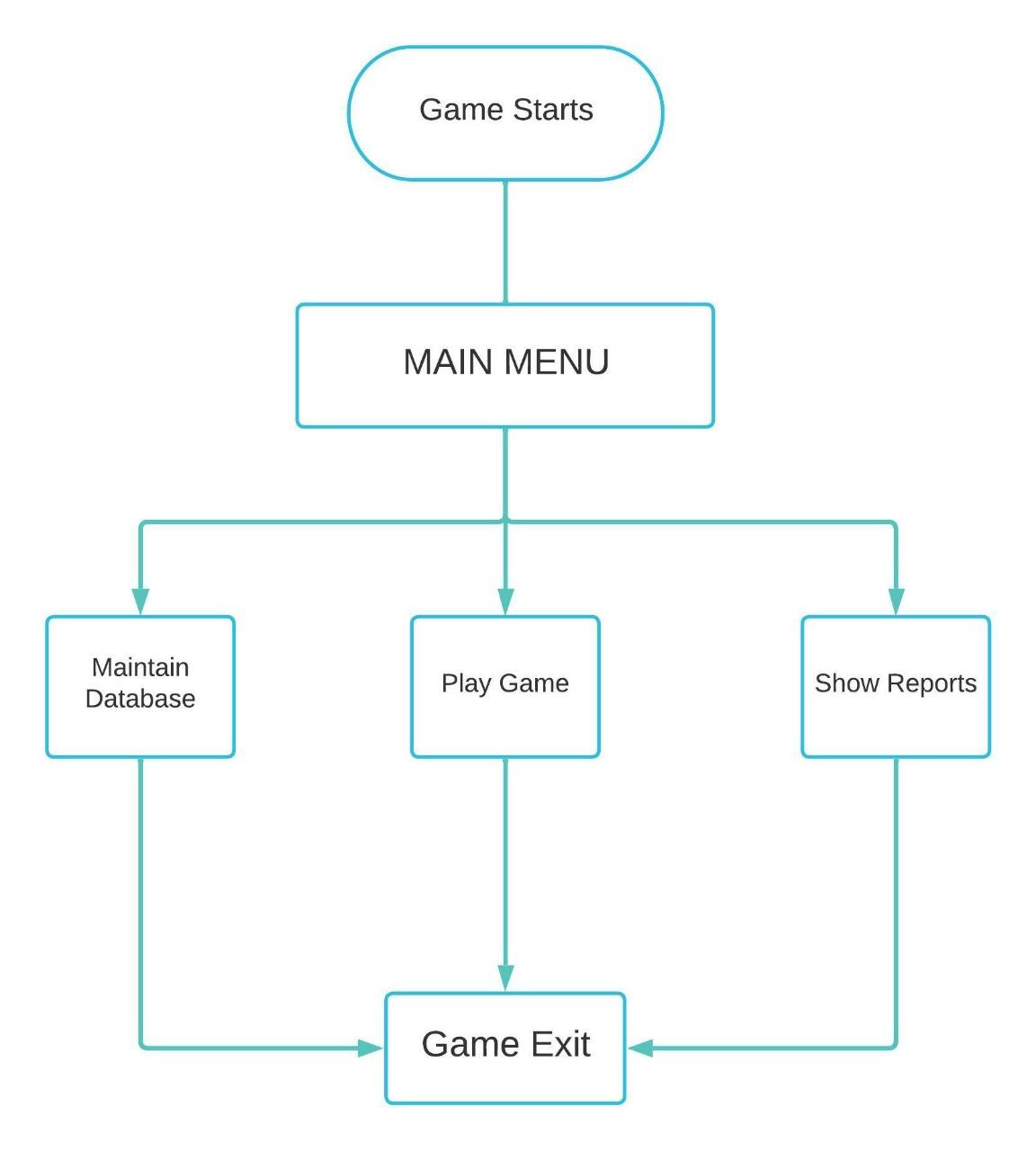
**4. DETAILED SYSTEM DESIGN:**

****

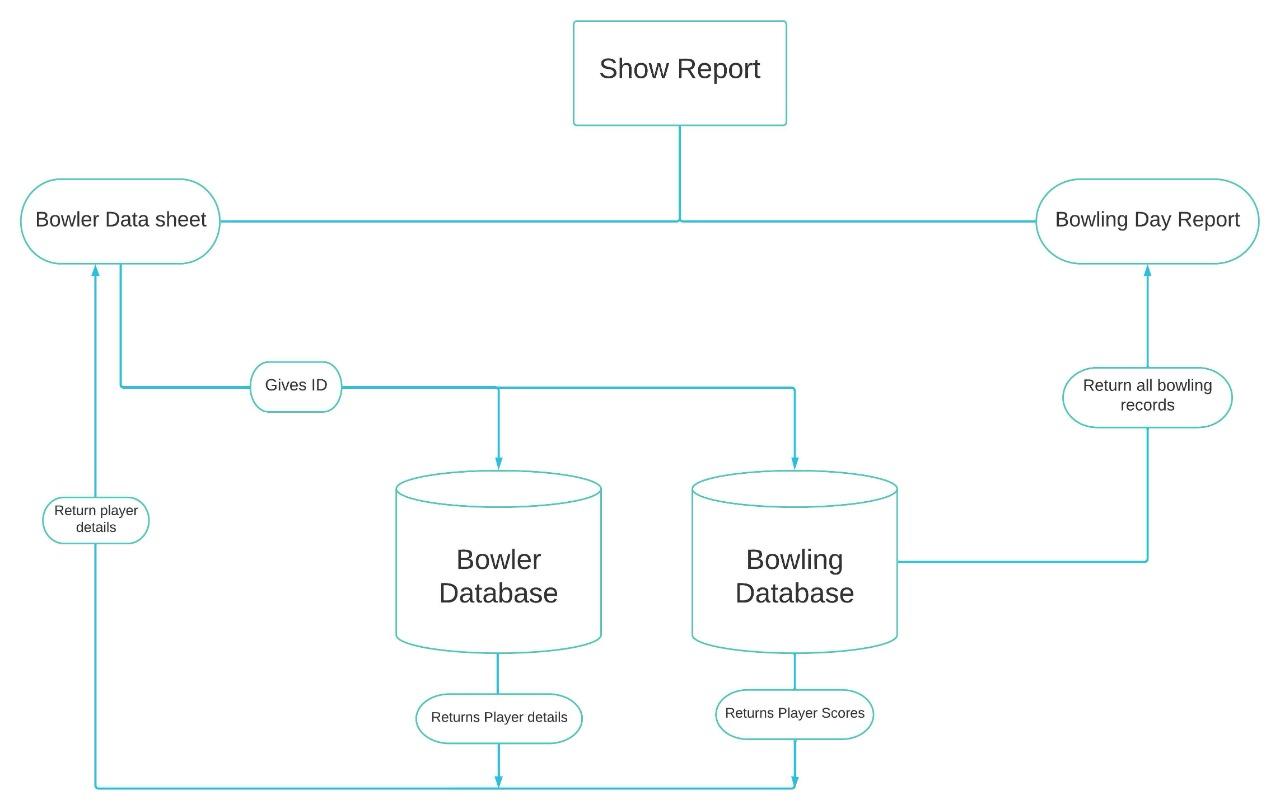
**4.1 Flow Chart of the application**

****

**4.2 Flow Chart for Maintain database function**



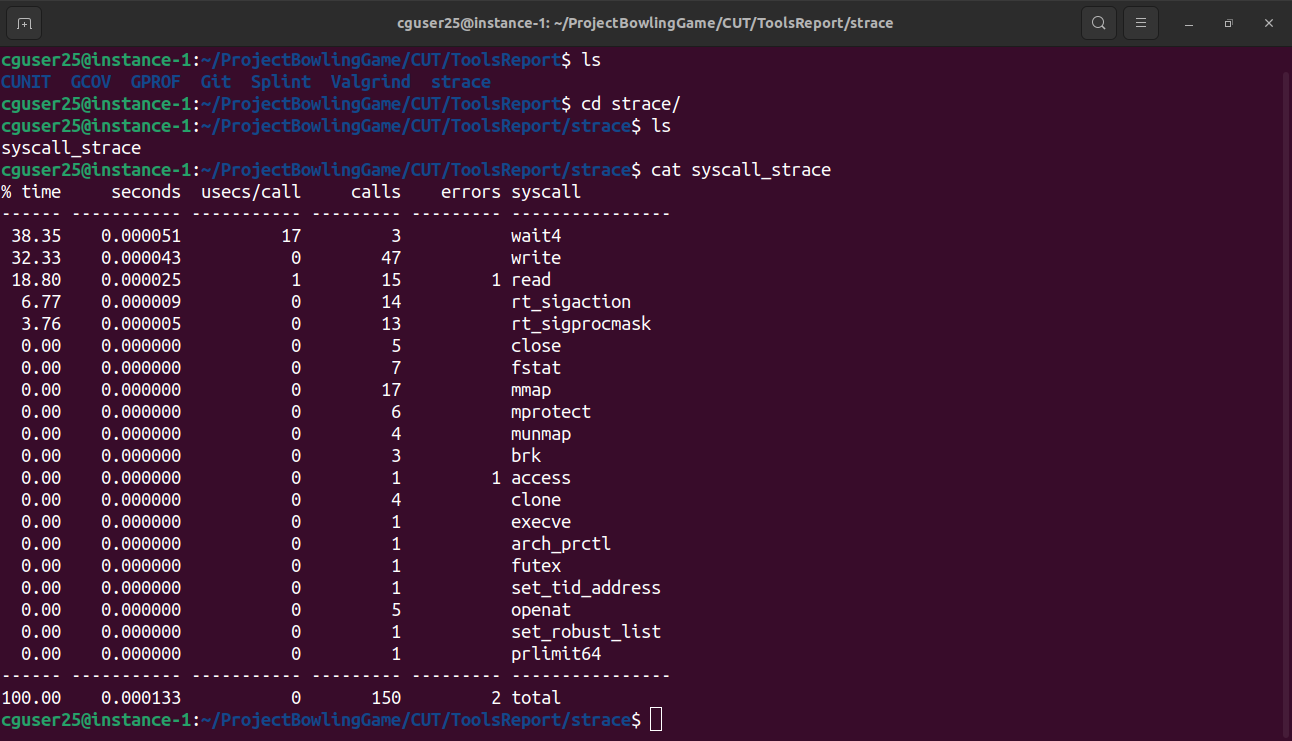
**4.3 Flow chart for Main menu of the application**

****

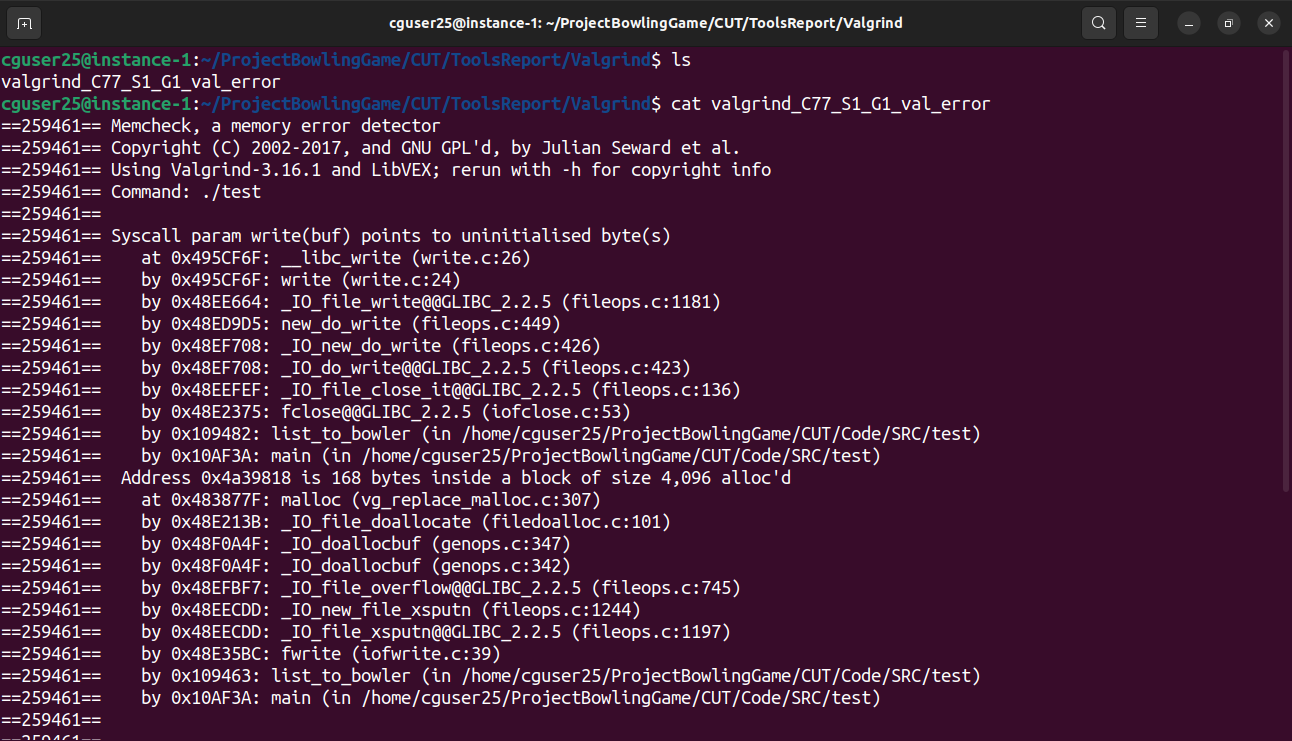
**4.4 Flow chart for Show report of the application**

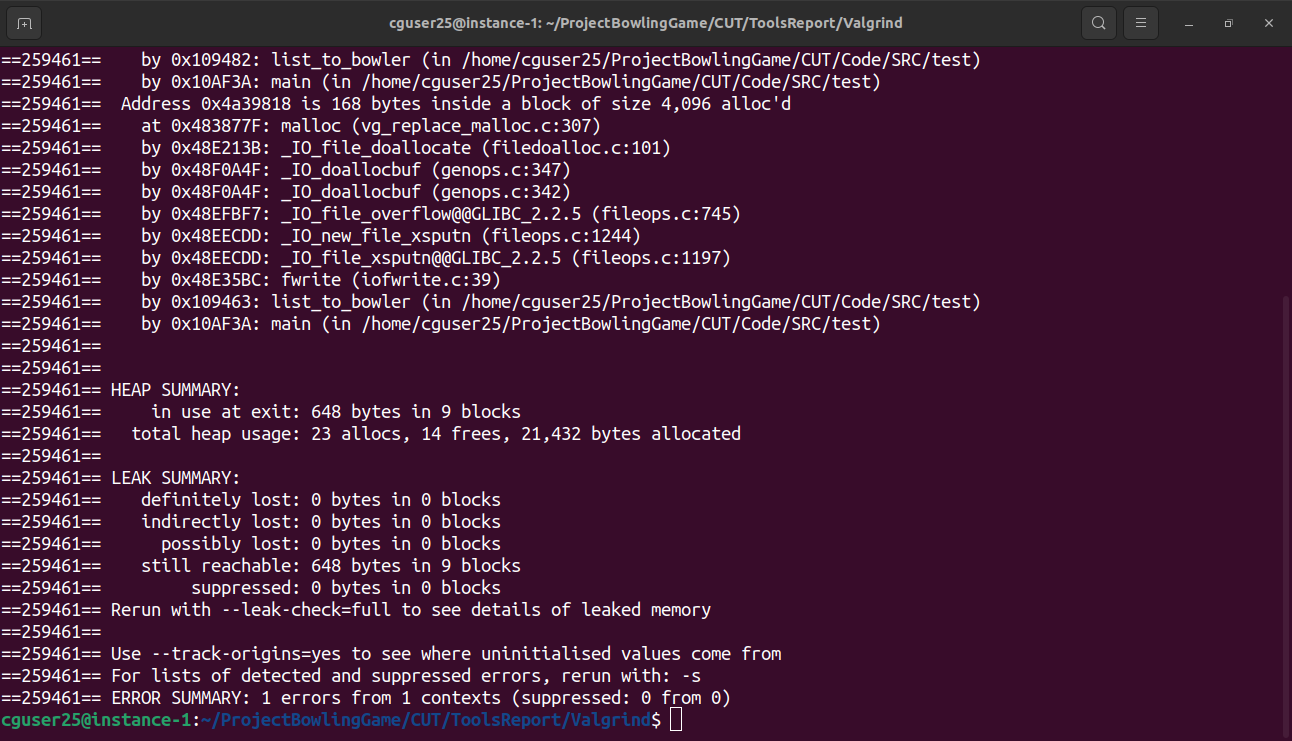
**5. TOOLS REPORT**

**5.1 Strace**

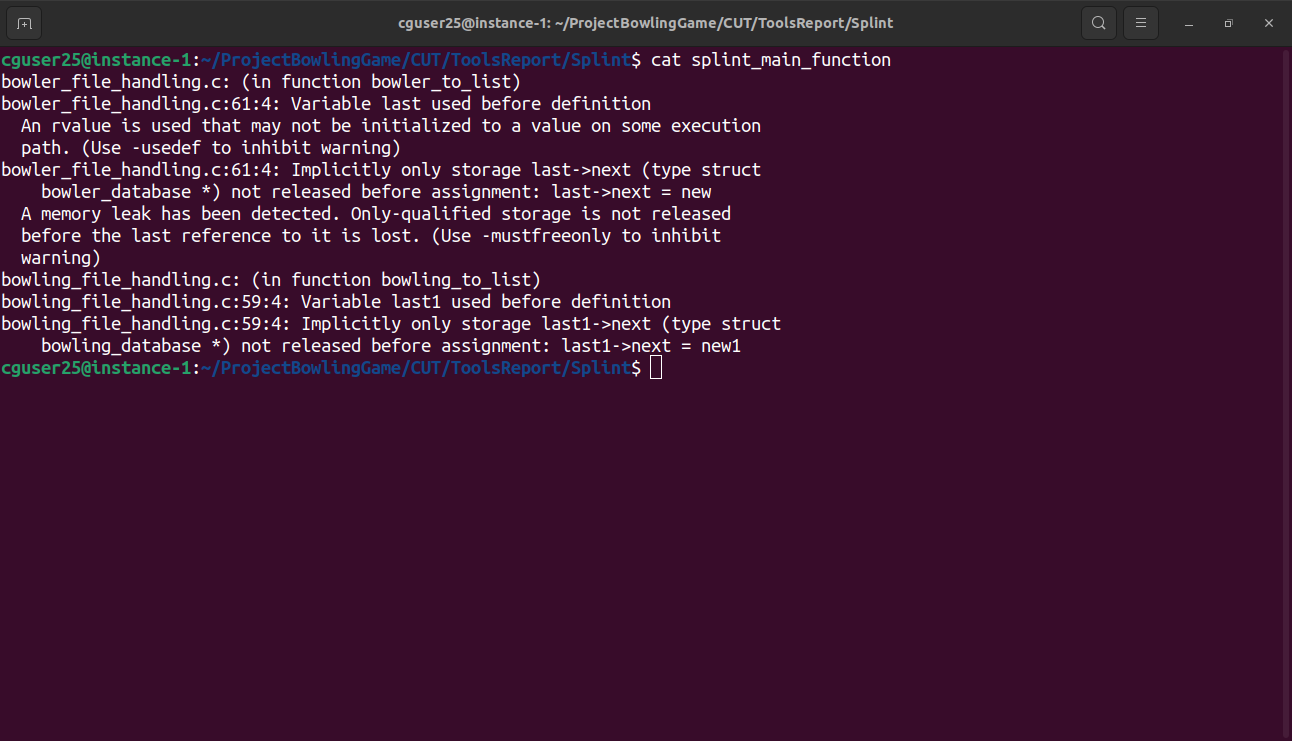
****

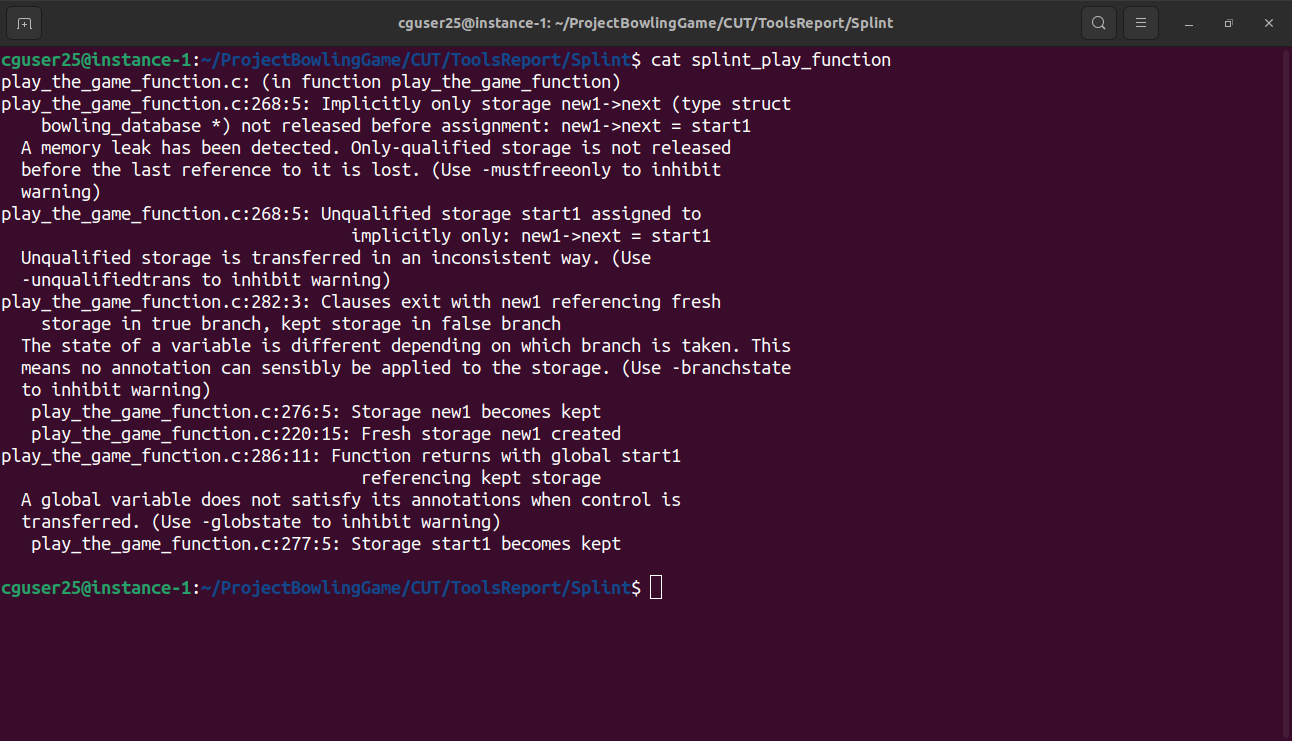
**5.2 Valgrind**

****

****

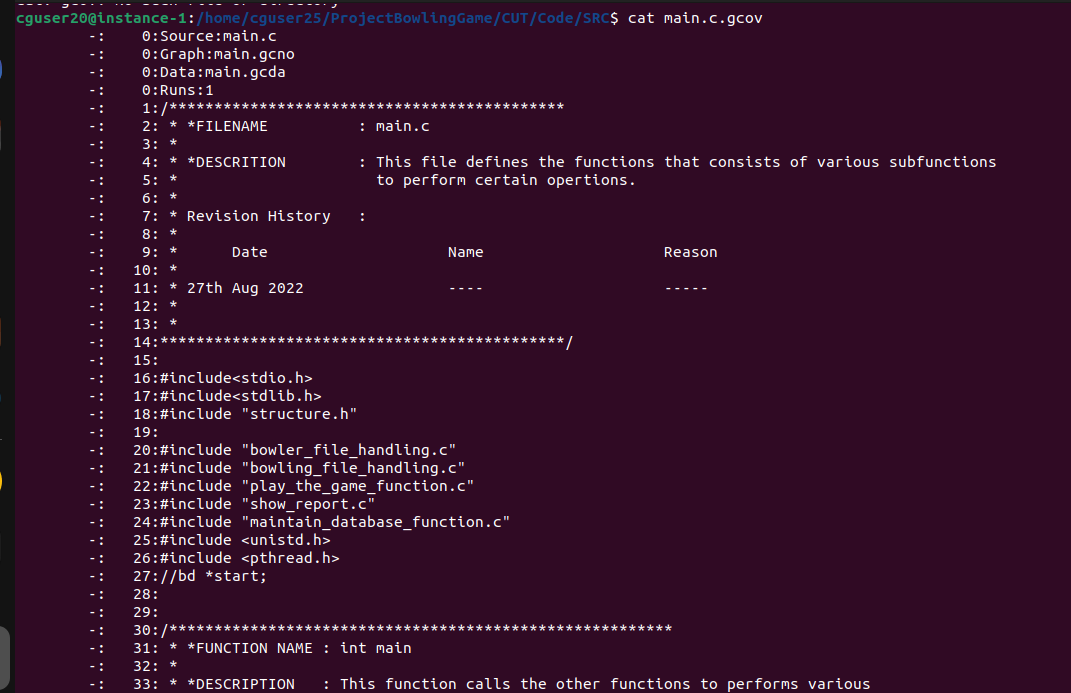
**5.3 Splint**

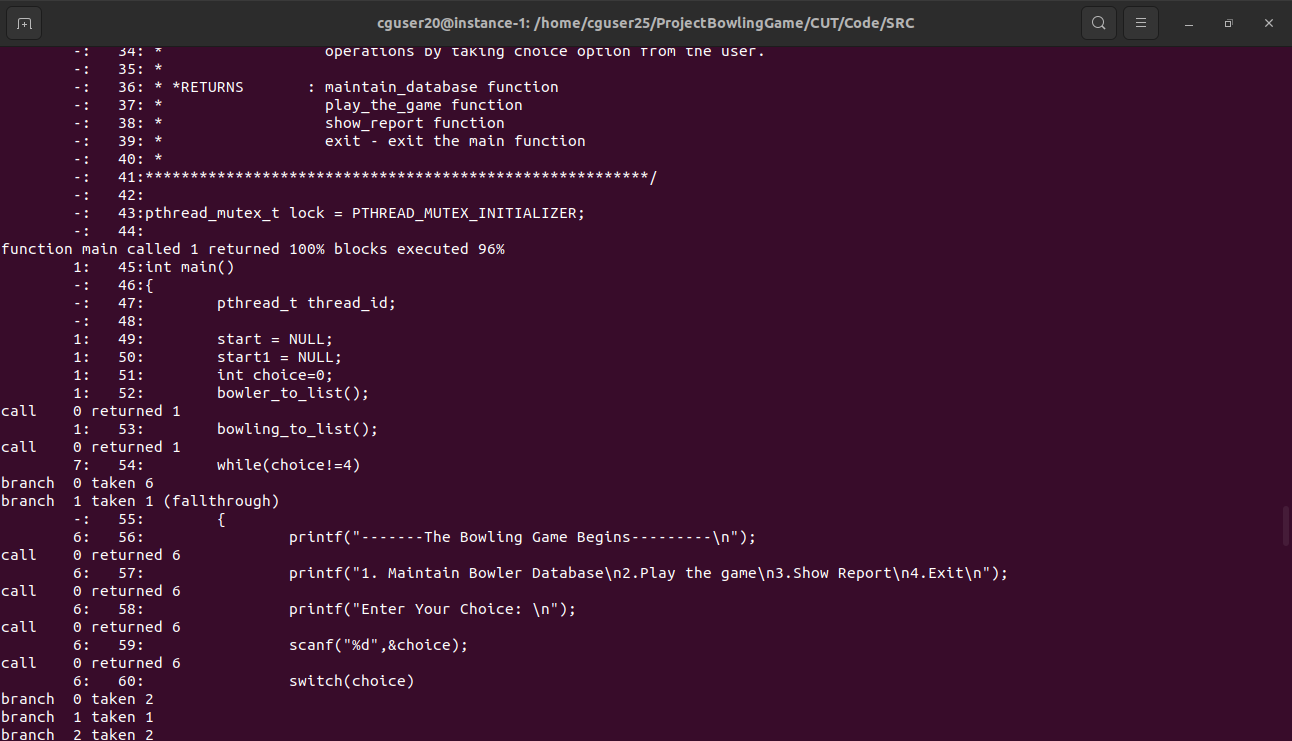
****

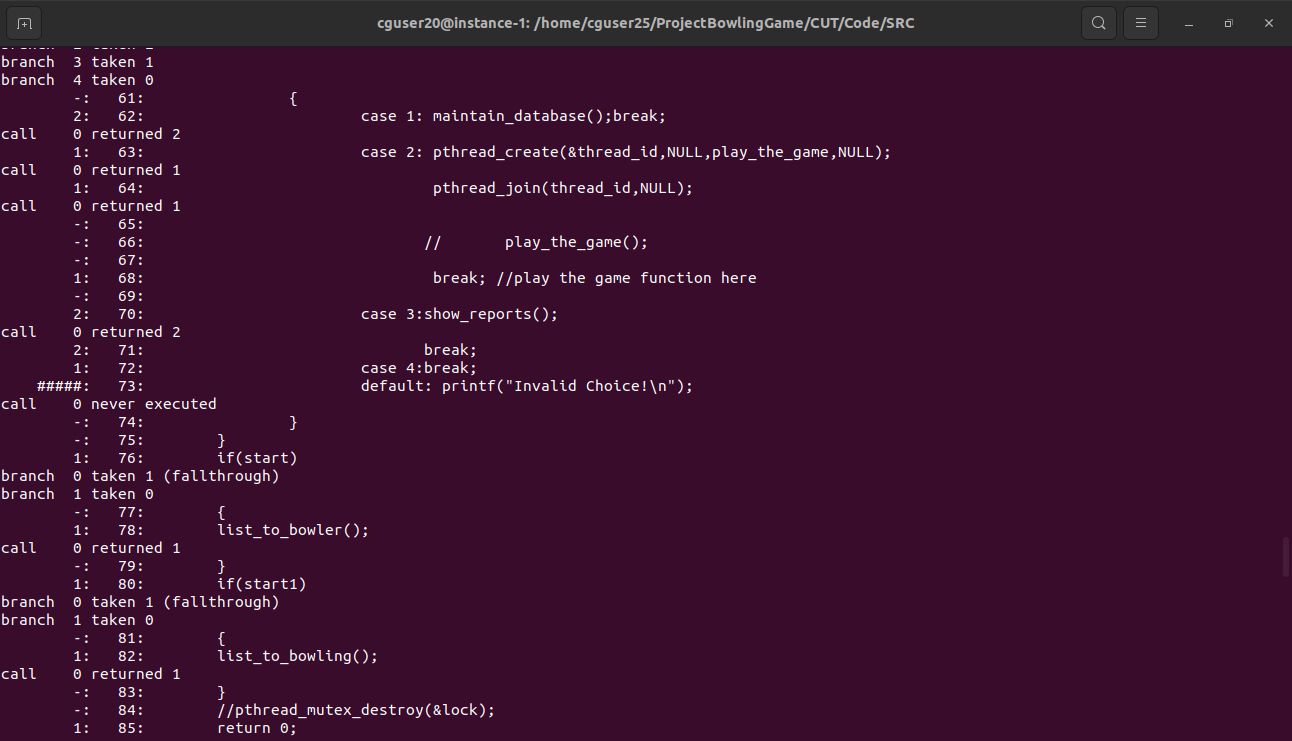
****

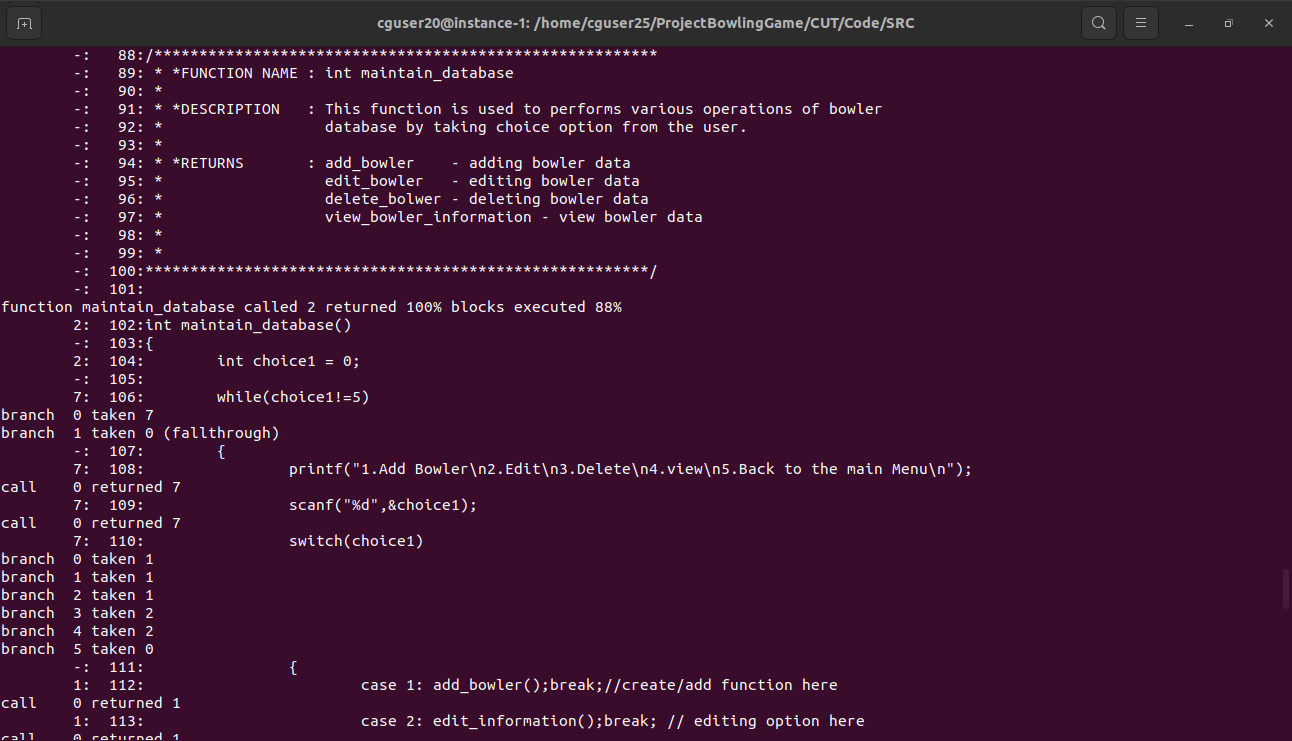
**5.4 Gcov**

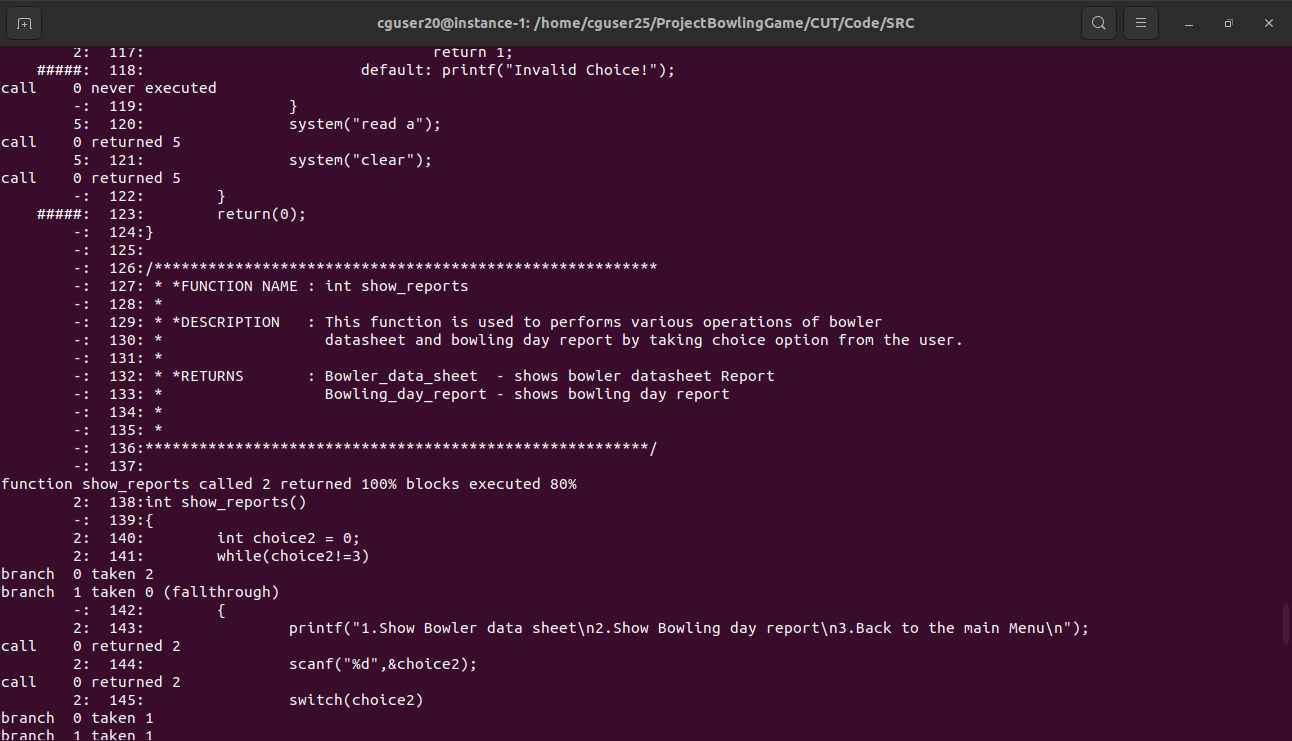
**Main.c.gcov:**

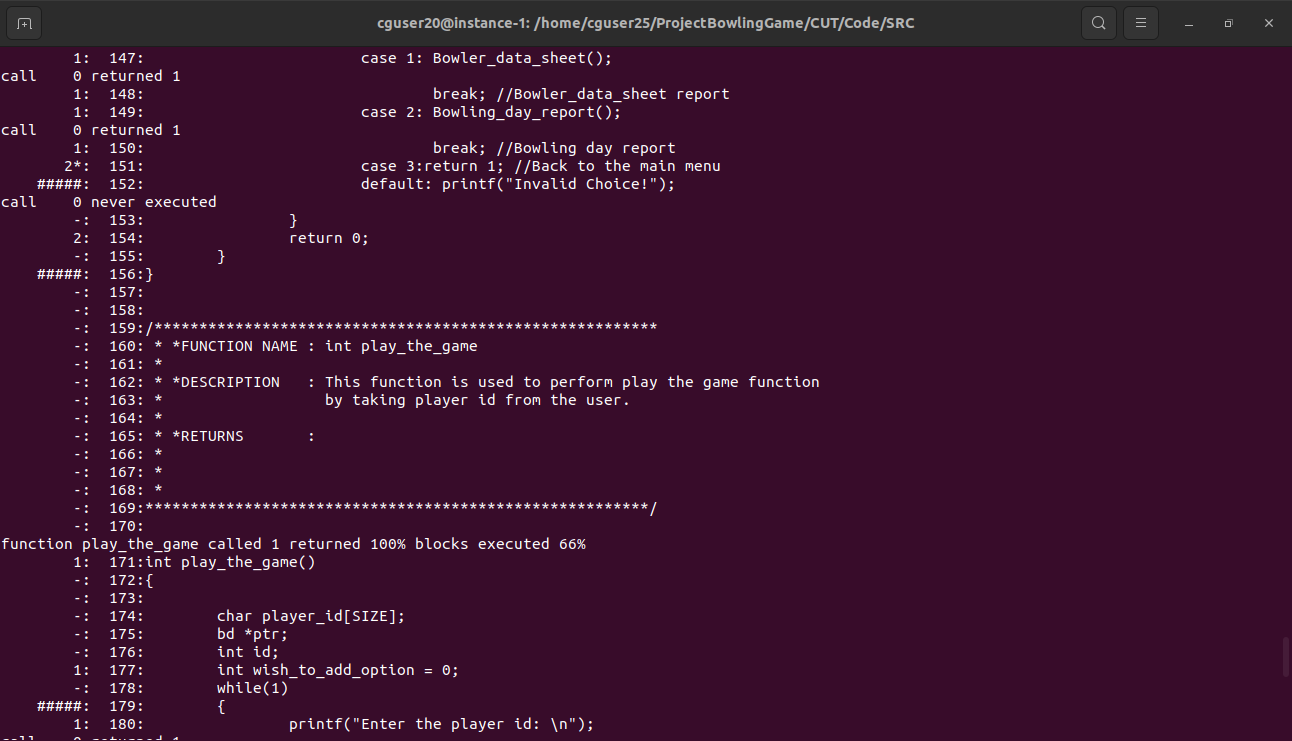
****

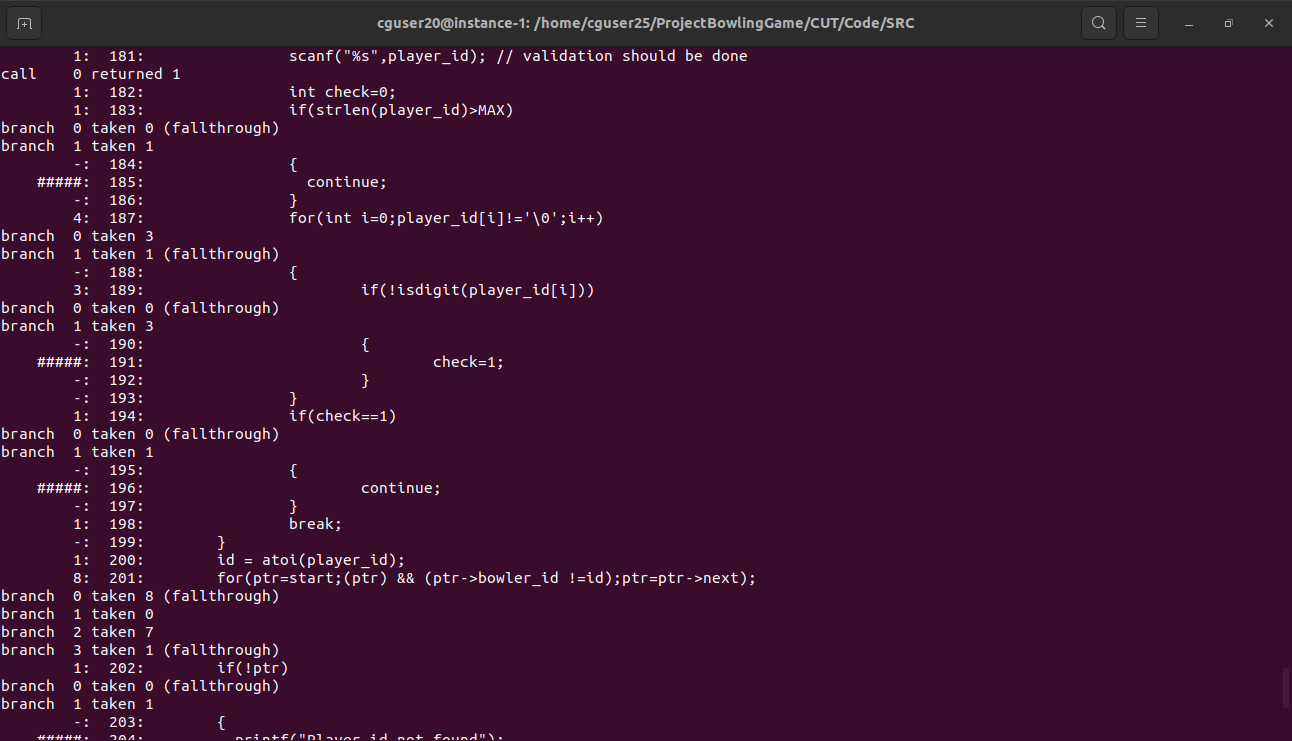
****

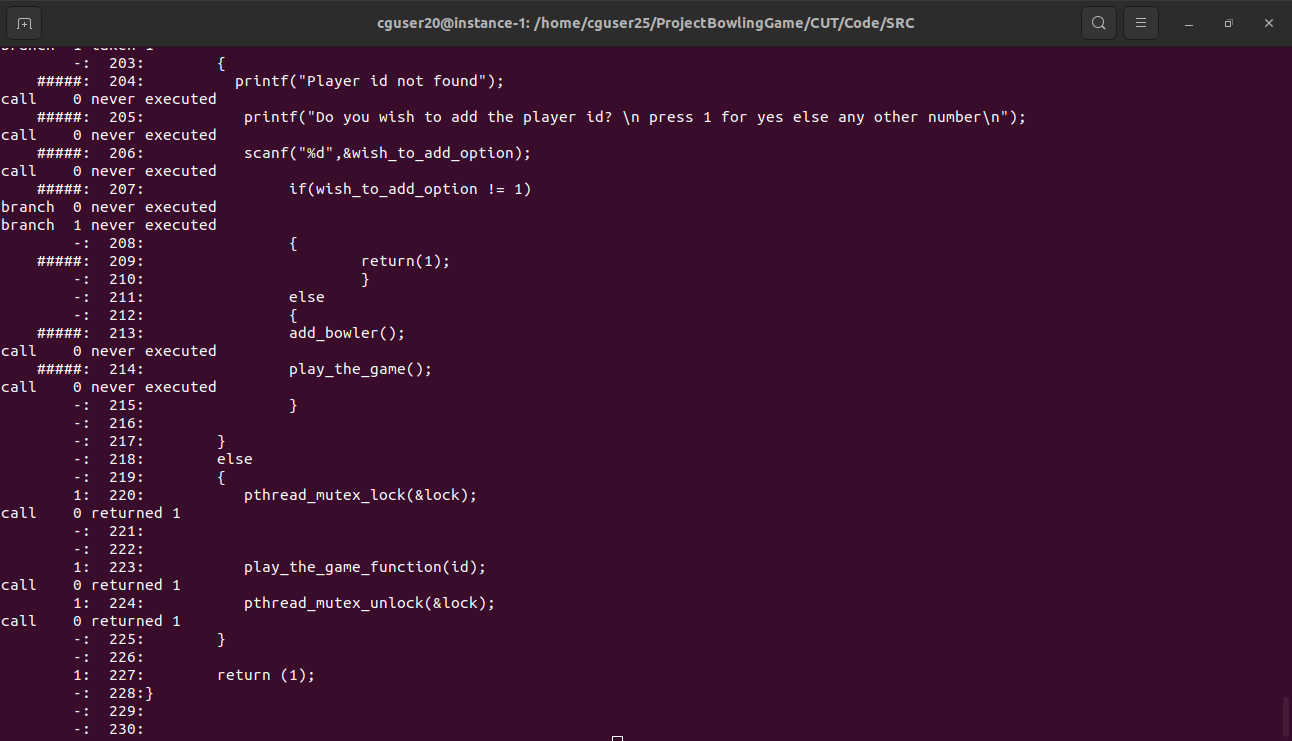
****

****

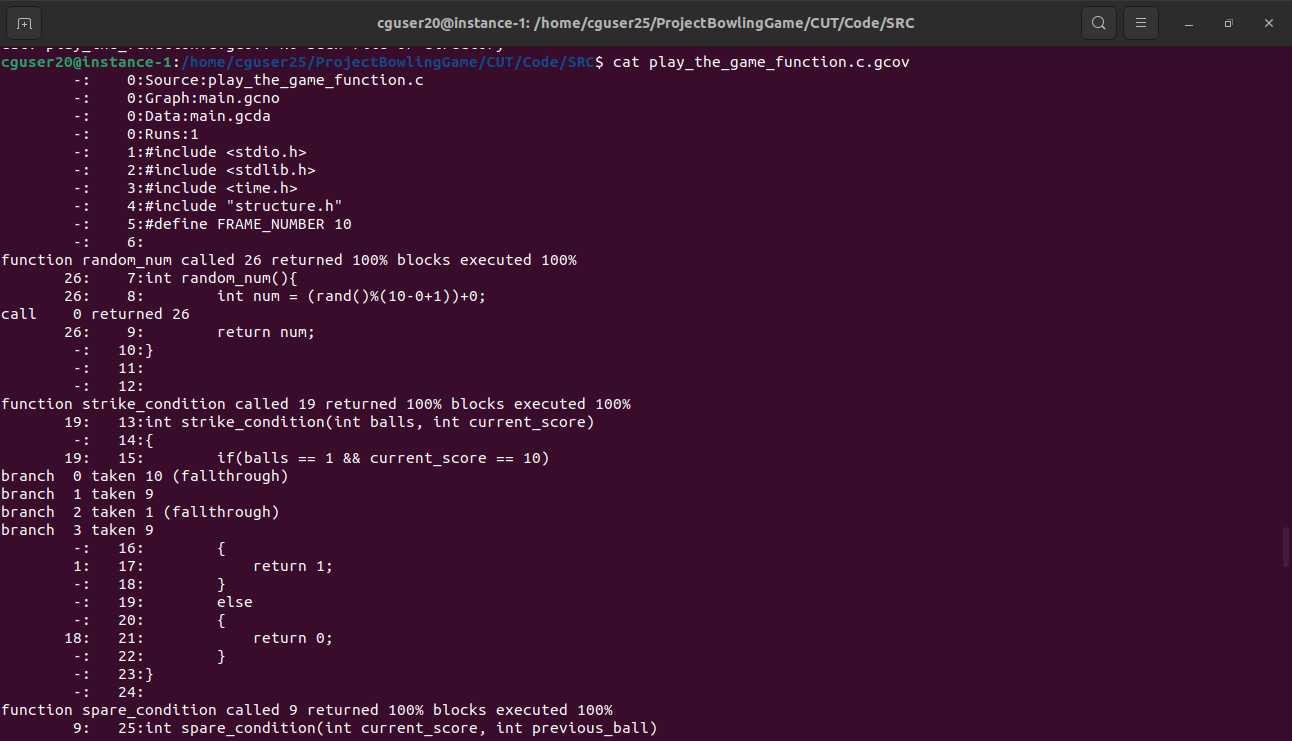
****

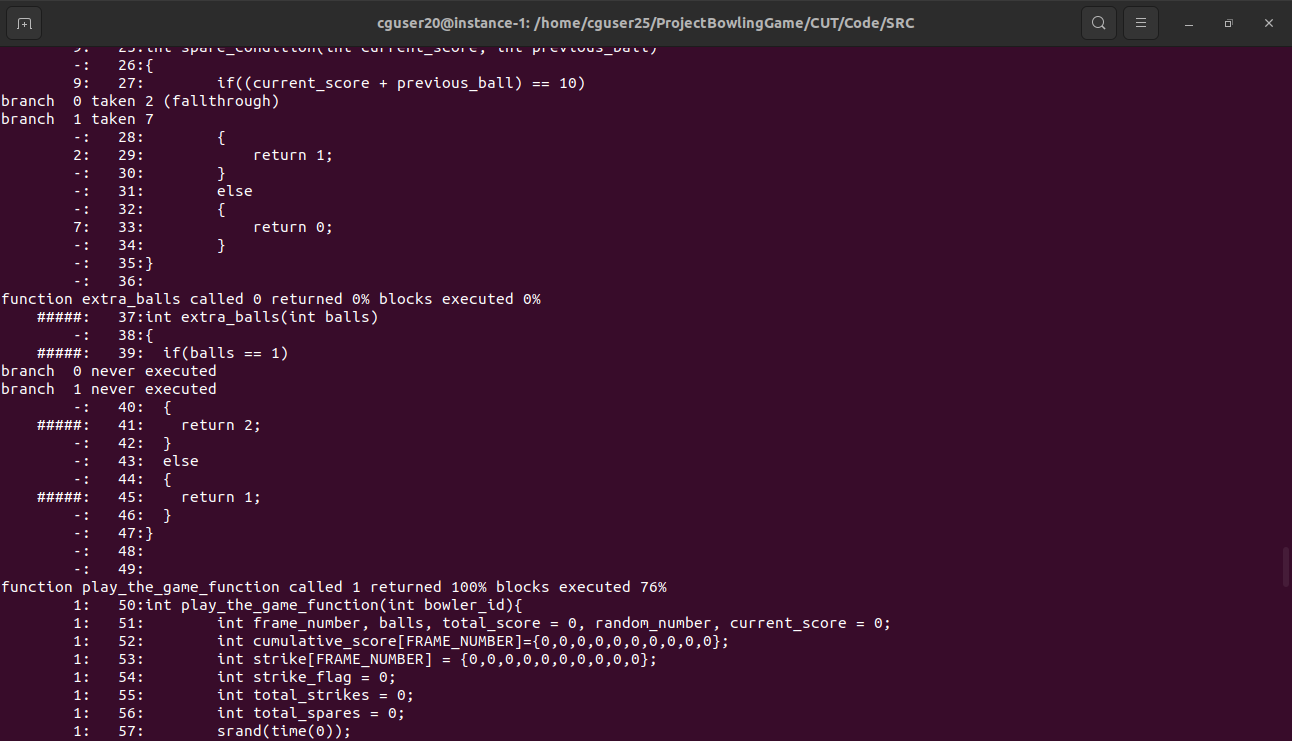
****

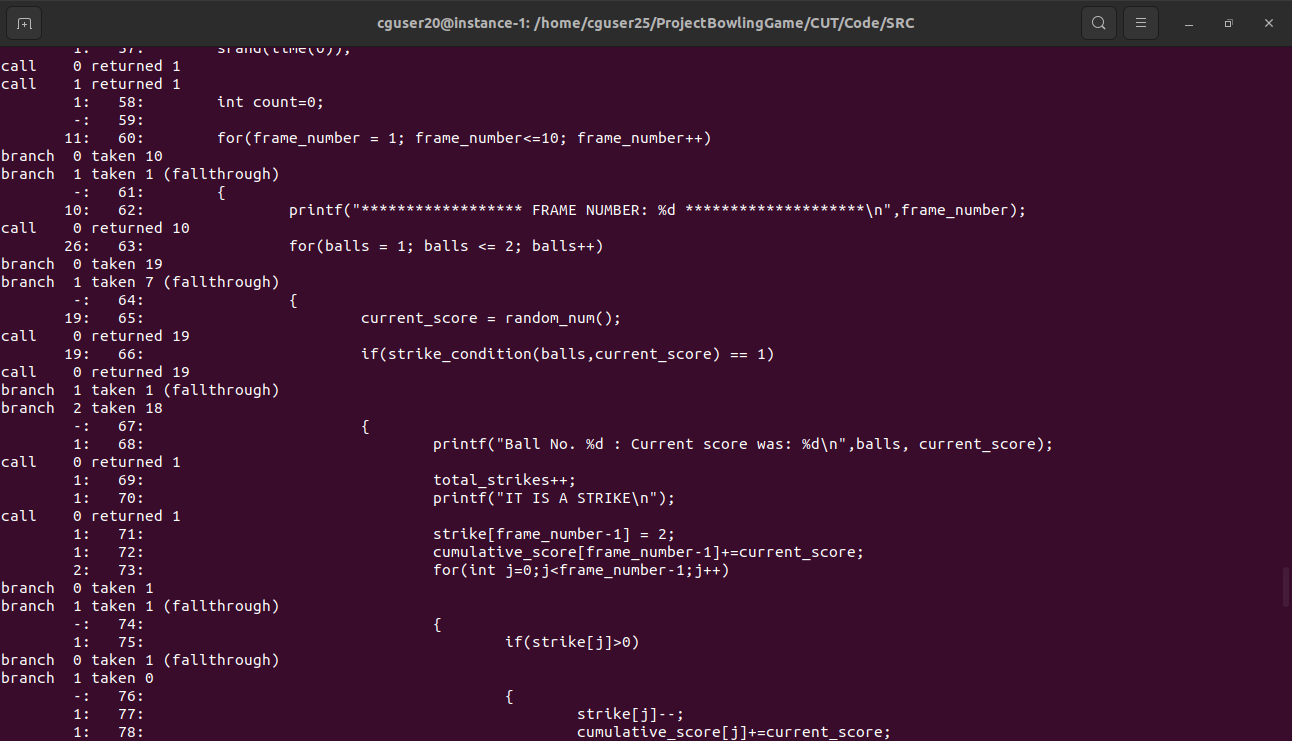
****

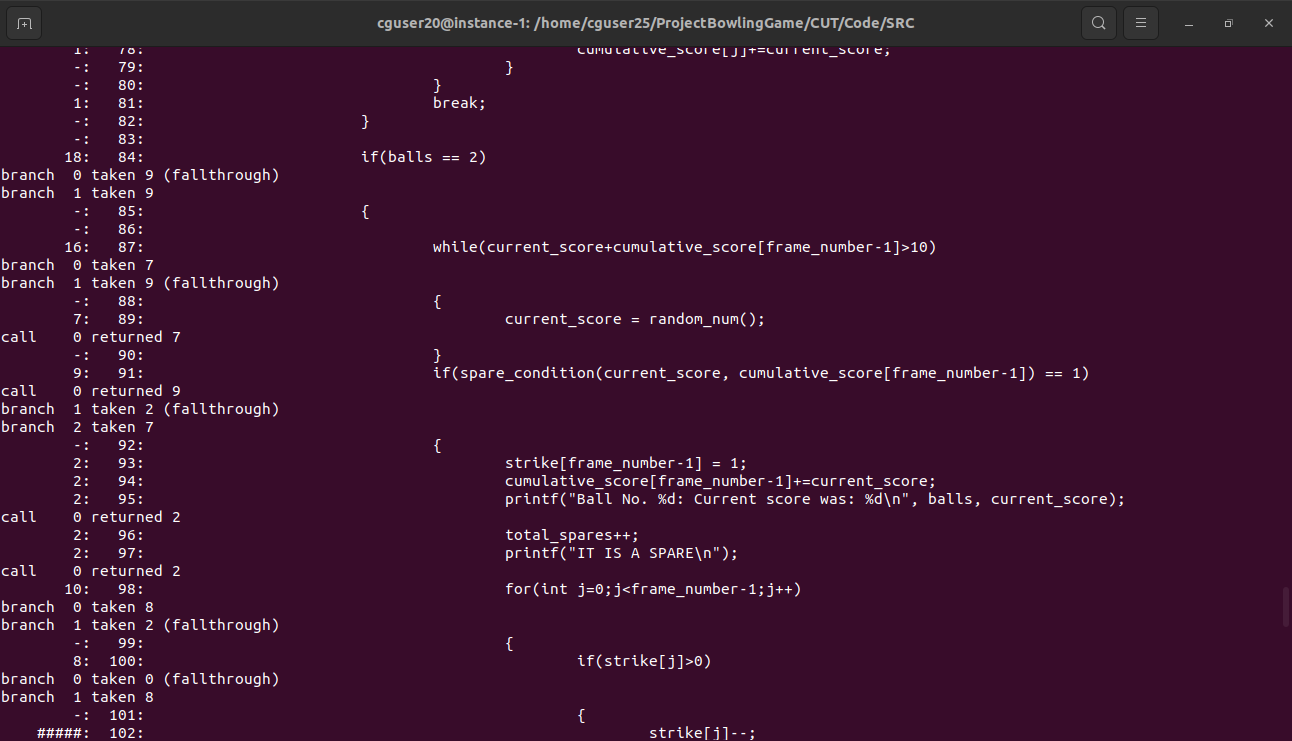
****

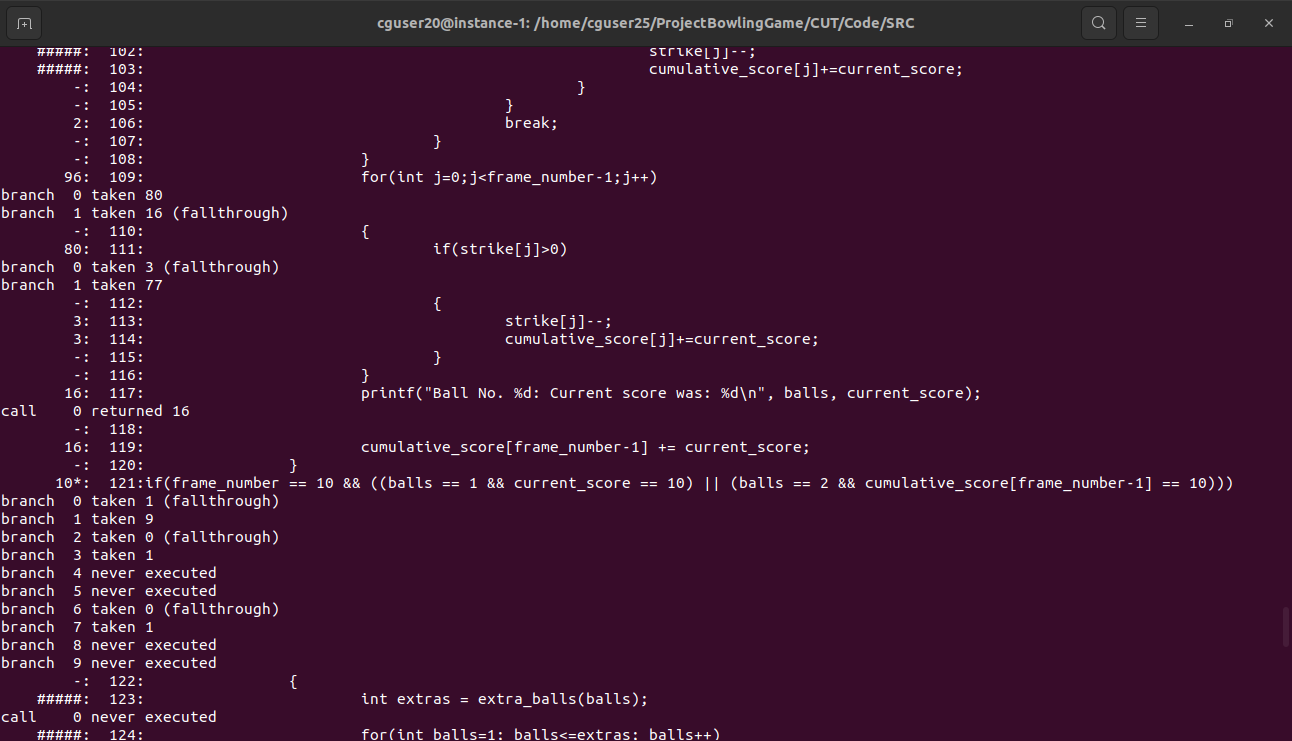
**Play\_the\_function.c.gcov**

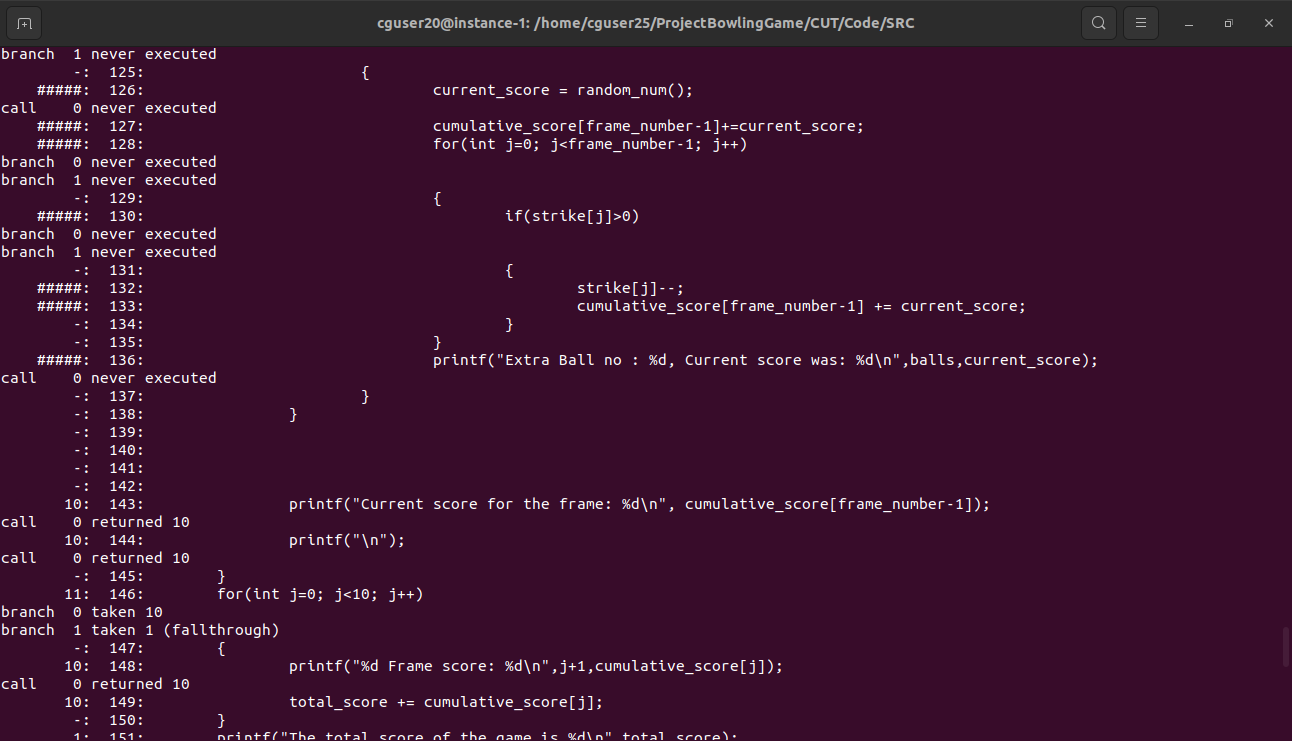
****

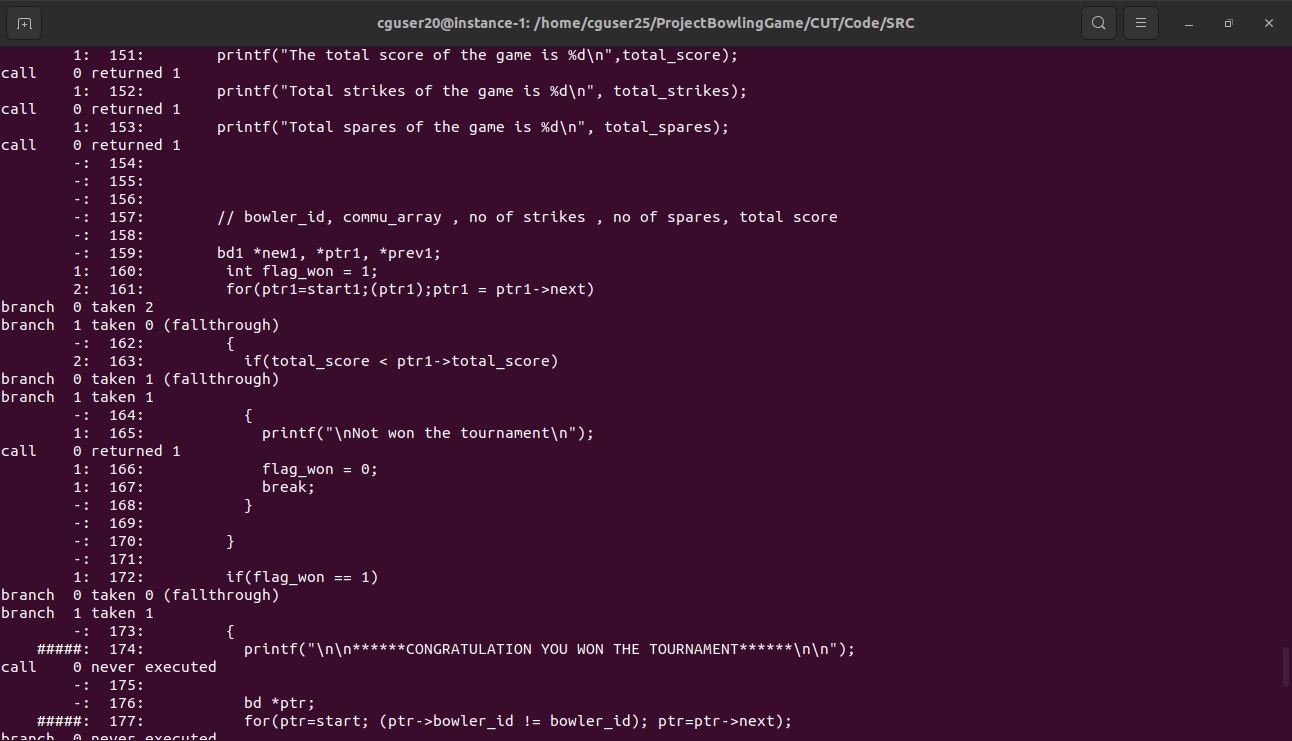
****

****

****

****

****

****

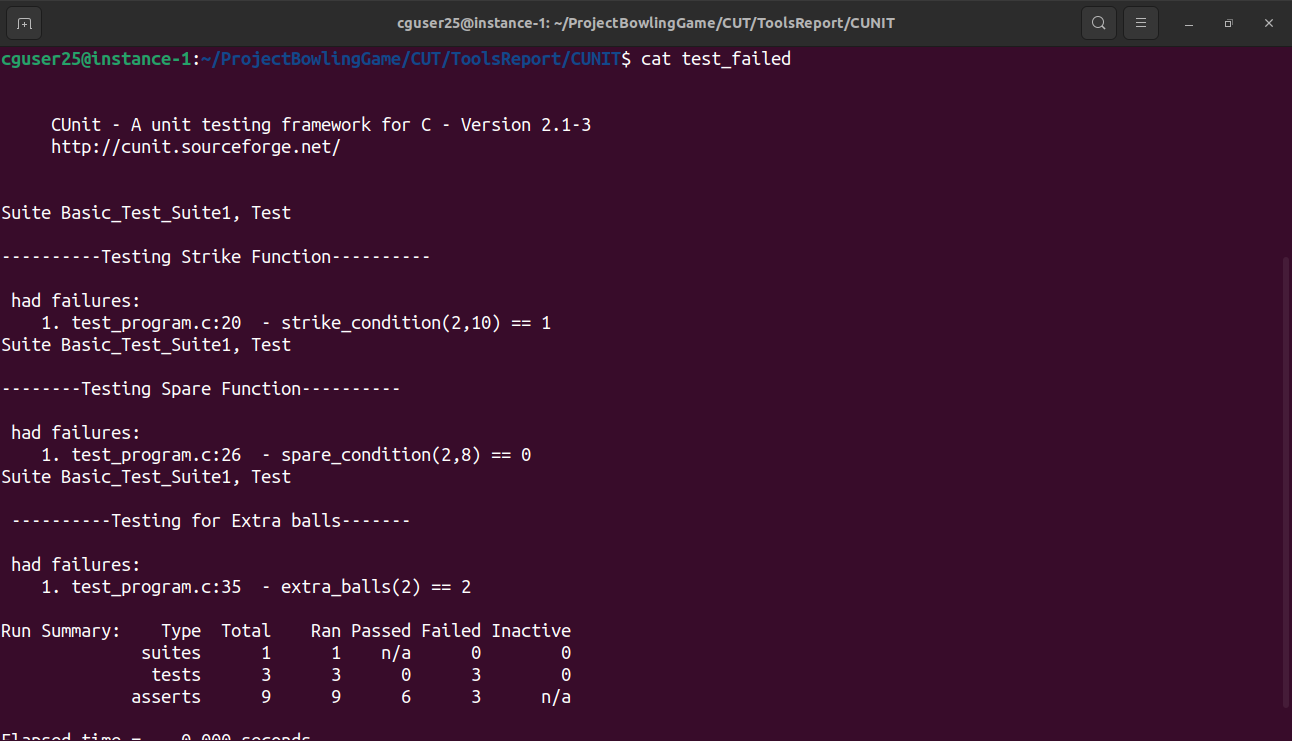
****

****

****

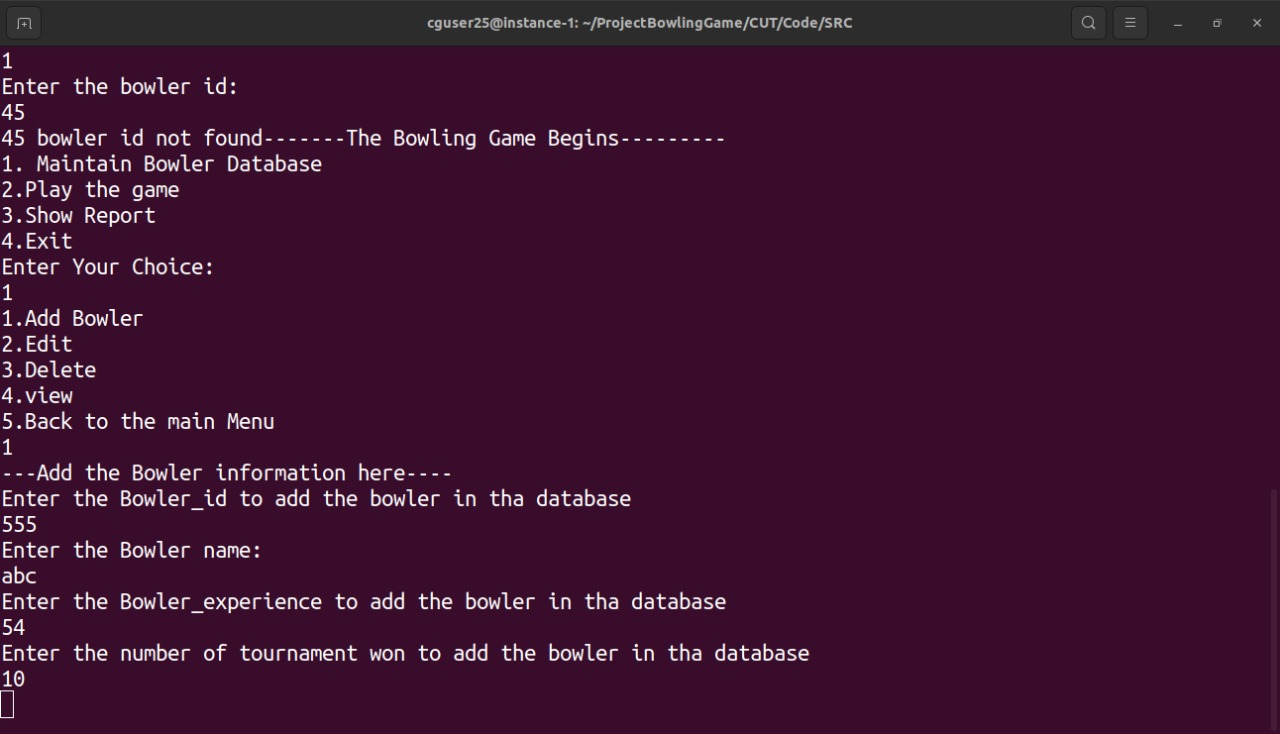
**6. Testing**

**6.1 Unit Testing**

****

**6.2 Integration Testing**

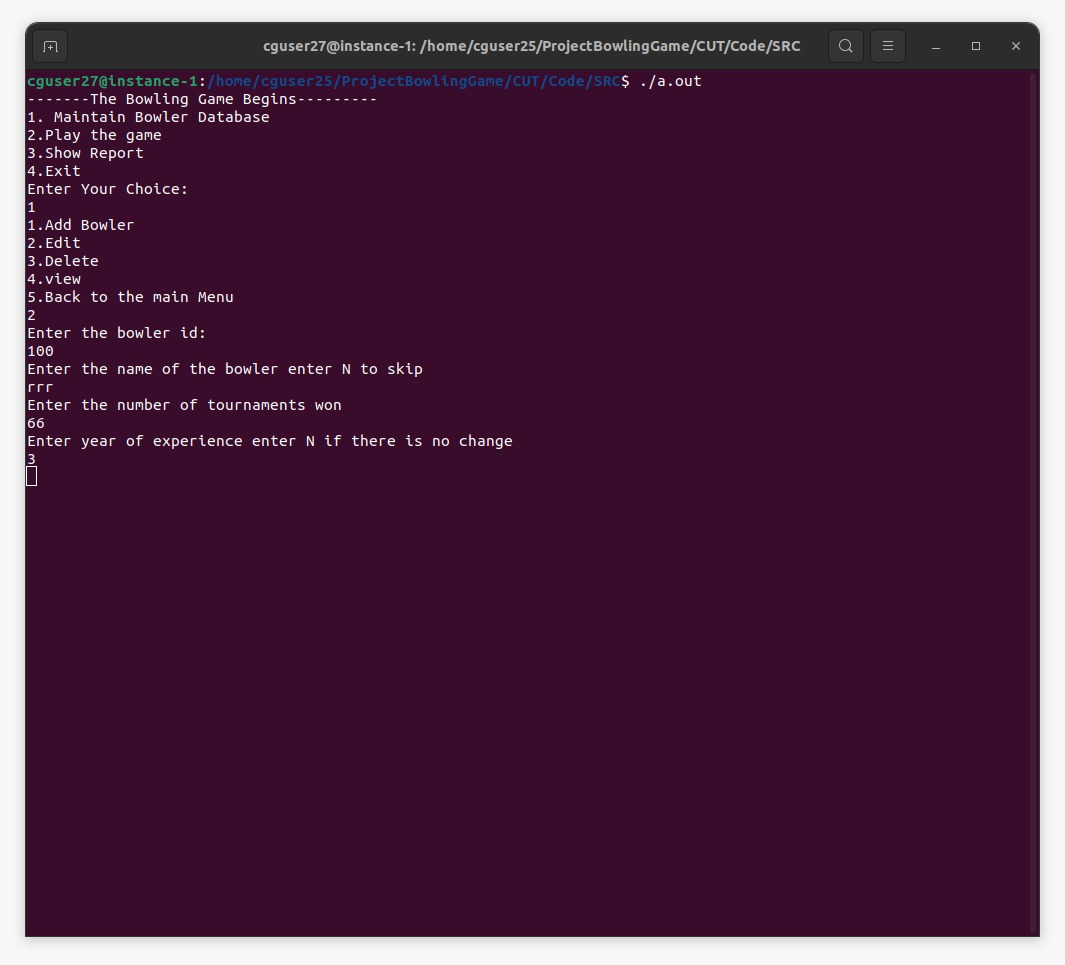
**6.2.1 Add**

****

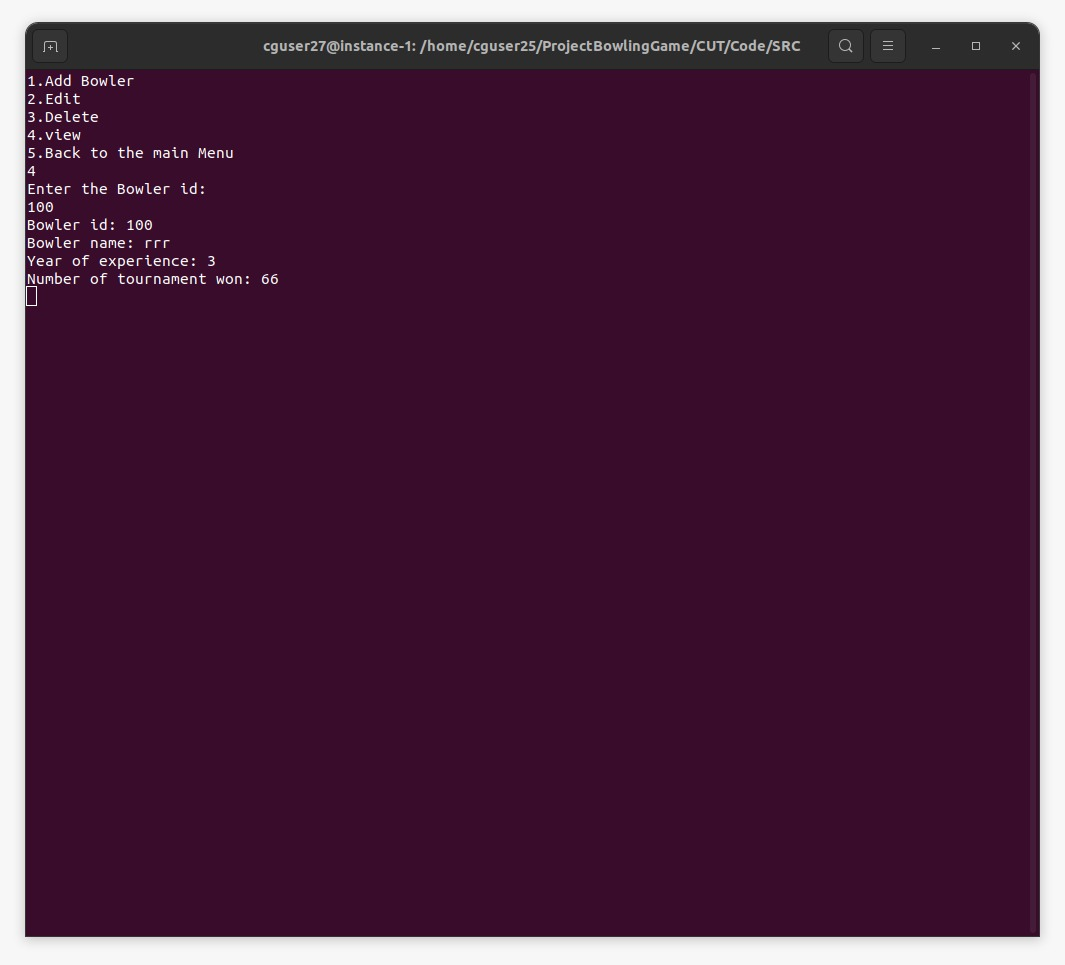
**6.2.2 Delete**

****

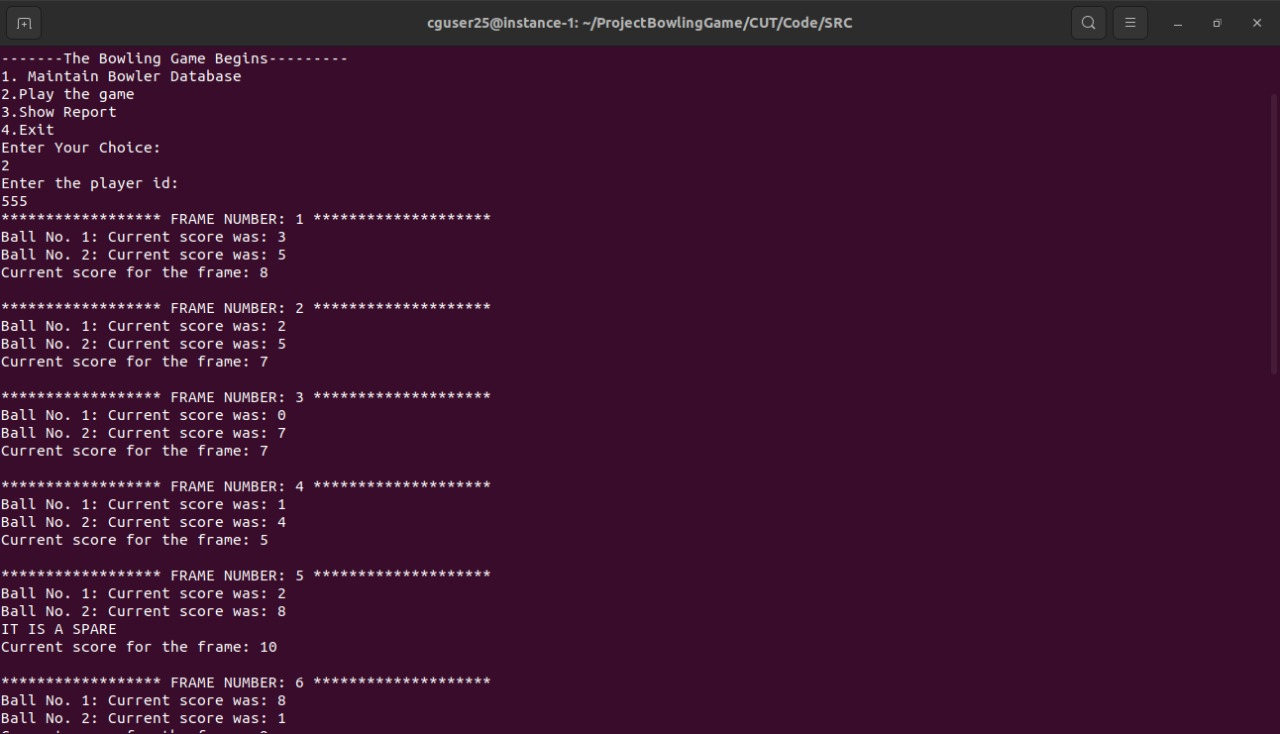
**6.2.3 Edit**

****

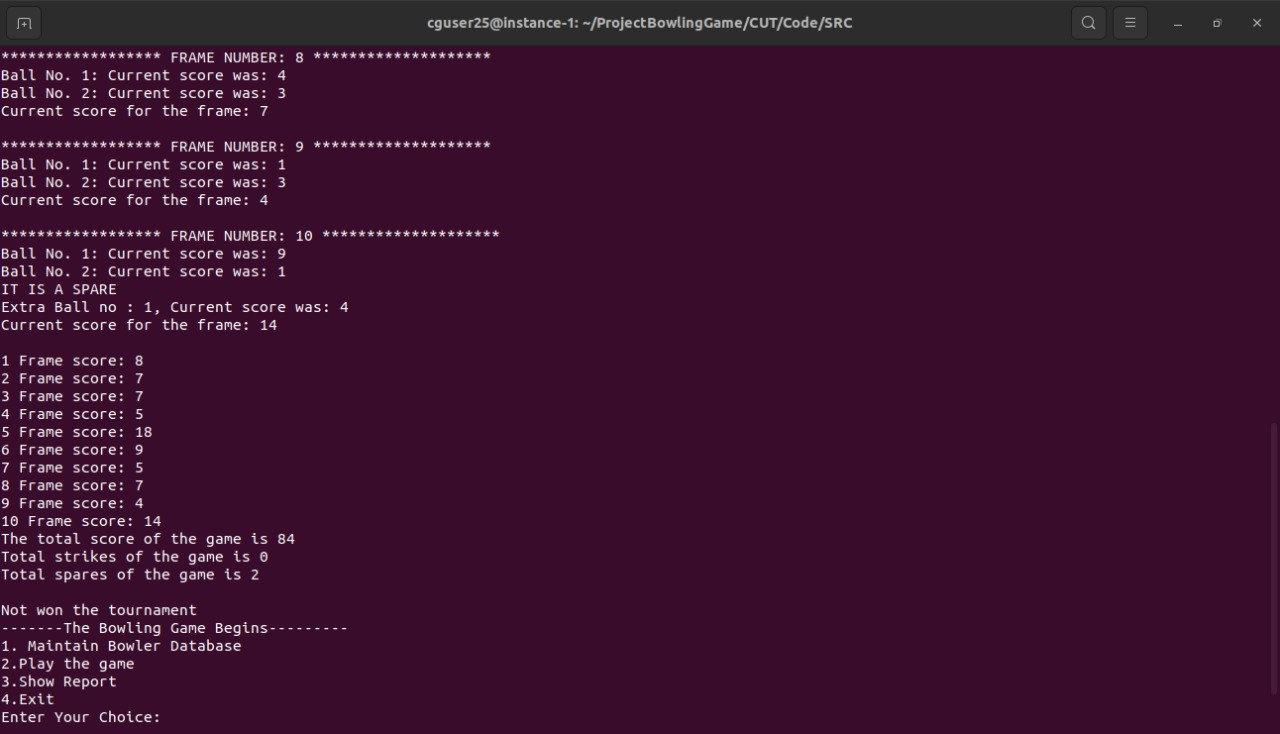
**6.2.4 View**

****

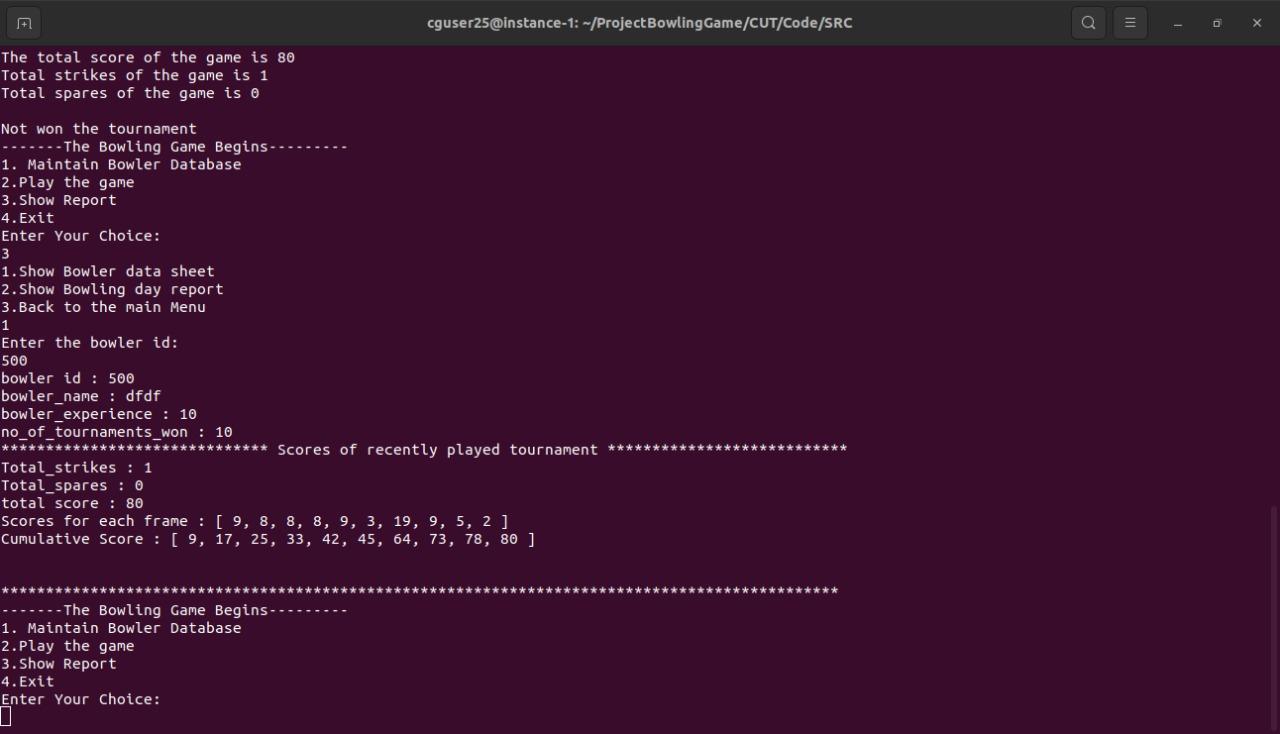
**6.2.5 Play the game**

****

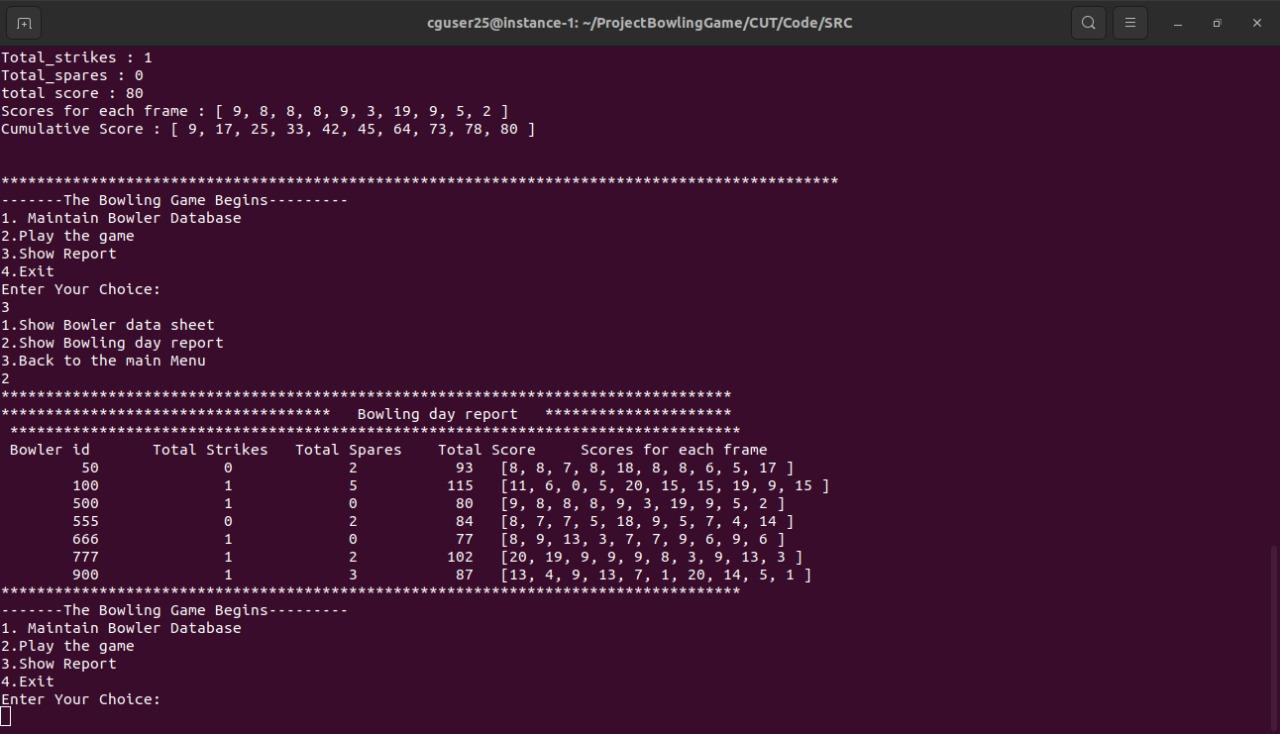
****

****

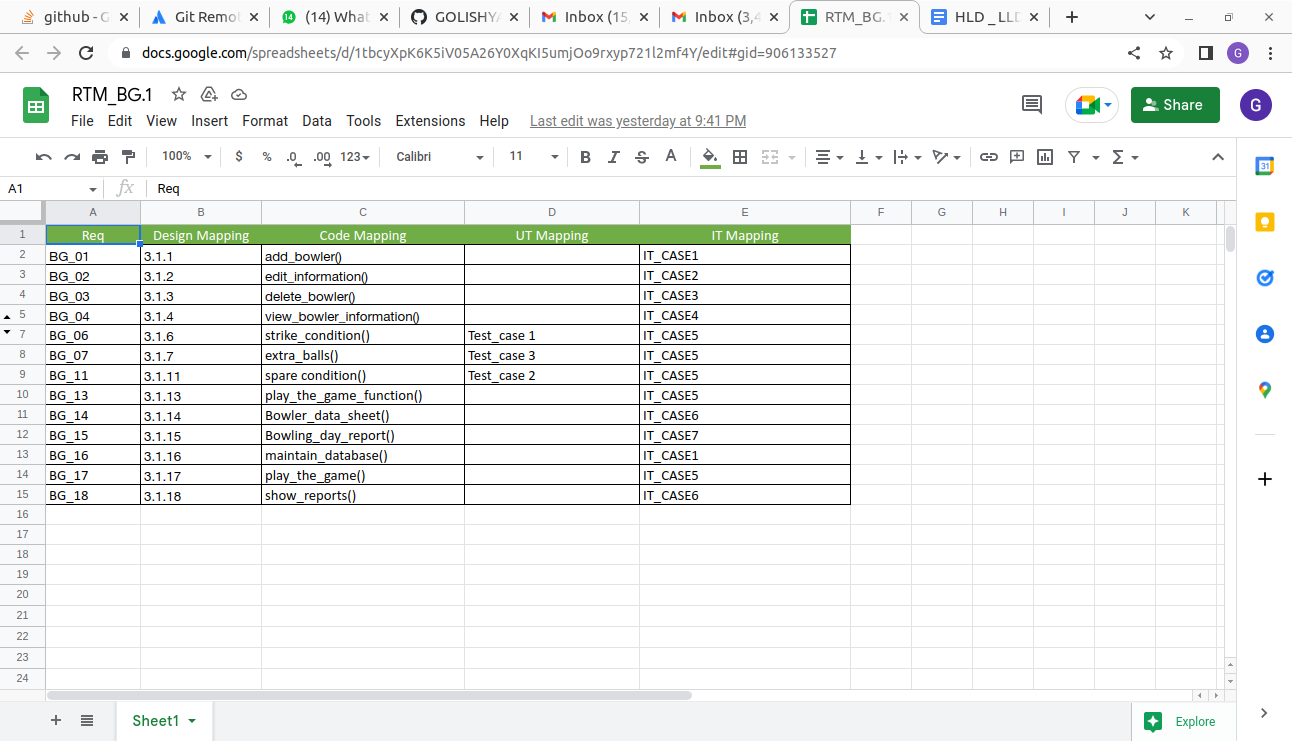
**6.2.6 Bowler data sheet**

****

**6.2.7 Bowler day report**

****

**7. Requirements Traceability Matrix(RTM)**

****